III. REMARKS

Claims 1-8 are not anticipated by Jackowski under 35 U.S.C. §102(e).

Claims 1, 4, 5 and 7 of Applicant's invention recited providing a uniquely identifiable identifier (UID, Stream Type) to at least one of the application and the data stream from or to the application, determining an association between said identifier and a particular QoS policy in a database stored in said terminal device, determining in said terminal device the QoS parameters contained in the QoS policy, and communicating from said terminal device to the network the QoS parameters to be applied to said at least one of the application and the data stream from or to the application. These features are not disclosed or suggested by Jackowski.

Jackowski discloses a system for policing traffic in an IP network, which comprises a number of workstations, an edge server and a policy server. In the system, an extensible service provider interface is placed between the winsock-API used by an application and the IP protocol stack in any of the workstations (column 7, row 58 - column 8, row 6). The workstations may be client or server type workstations. To the extensible service provider interface may be attached a number of plugin programs. One such plugin program is an application-classifier plugin (ACE) (column 8, rows 23-56). The application-classifier plugin collects network statistics for packets originating from or received by a workstation. The application-classified plugin program provides an event notification service for a controller application (column 9, rows 37-49). The application-classifier plugin provides the controller with notifications on events such

as the starting and stopping of applications, the opening of sockets, the connecting of sockets, the writing and reading of sockets, and the closing of sockets. The format of the event notifications is illustrated in Figures 7A-7E. The controller is also informed of a process identifier. The application-classified plugin obtains the name of the application associated with the process identifier (column 10, rows 63-67). The controller forms tables, the format of which is illustrated in Figures 9A and 9B.

The system taught by Jackowski comprises a separate policy server (column 15, rows 20-43), which collects statistics of network traffic from the workstations. The policy server is responsible for assigning priorities to flows originated or terminated by the workstations. The policy server is provided information on flows that are associated with given source and destination addresses and TCP-ports via the edge server (column 15, rows 27-33). The policy server acquires information on the applications that are associated with these flows and statistical information collected via the application-classifier plugin and the controller in a workstation (column 15, rows 23-35). The policy server compares the consumed capacity to acceptable threshold values. If the threshold values have been exceeded, the policy server instructs the edge server to regulate the traffic by delaying or dropping packets (column 15, rows 36-52).

In the solution taught by Jackowski, the QoS determination based on the application is made in an external policy server so that the connection from the workstation to a remote server is made first. The actual policing will be performed afterwards by comparing the traffic statistics associated with the application

to threshold value (column 15, rows 35-39). Further, the policy server acquires flow data from the workstations.

Jackowski does not disclose or suggest *determining an association between said identifier and a particular QoS policy in a database stored in said terminal device*. In Jackowski, the actual QoS is obtained only in the separate policy server (column 15, rows 35-39). Jackowski does disclose that the controller application in the workstation collects statistical information pertaining to a given application such as average, minimum and maximum data rates, but the actual comparing of these values to acceptable thresholds is performed in the policy server (column 15, rows 35 -39). The passages cited by the Examiner refer merely to the event notification formats. The statistics collected may not be pertinently referred to as a *QoS* policy.

Jackowski also does not disclose or suggest *determining in said terminal device QoS parameters contained in the QoS policy*, or that the QoS parameters are determined in the terminal device. As explained above, Jackowski teaches that acceptable threshold values are obtained only in the policy server. The passages cited by the Examiner for this feature only teach that the policy server enquires the traffic statistics from the workstations and may instruct an edge device to block or delay packet for flows originating from certain applications (column 15, rows 14-62). As to the question of whether the term "terminal" applies pertinently to the policy server, the feature that a *certain Quality of Service (QoS) is applied to a data stream of an application executing in a terminal device*, recited in the preamble must be considered. In the light of this feature the policy server is not a terminal.

Furthermore, Jackowski fails to disclose or suggest *communicating from said terminal device to the network the QoS parameters to be applied to said at least one of the application and the data stream from or to the application.* In Jackowski, the terminal does not communicate the QoS parameters, which are to be applied for an application or a data stream of a given application, to the network. As to the policy server being the terminal, the considerations stated above apply also to the this feature, and the policy server is not a terminal.

Additionally, Jackowski fails to disclose or suggest of *providing a uniquely identifiable identifier* (UID, Stream Type) *to at least one of the application and the data stream from or to the application.* In Jackowski, the unique identifier is a process identifier (Figure 7B). As a person of skill in the art will appreciate, a process identifier does not reveal the actual application executing as that process. For example, there may be several instances of an application executing in parallel. Each has a unique process identifier, which does not reveal the actual underlying application. The application name must be obtained separately using the process identifier. A process identifier may be later on assigned to an instance of a different application. Further, the lack of a uniquely identifiable identifier is also evident from Jackowski (column 15, lines 59-63), where low-priority web browsing from the client can be identified by finding the application name in the flow tables for the IP address for client and port used by the browser. However an application name is not a uniquely identifiable identifier, because the application name is unique only with a certain probability. The implementation relies on the fact that rarely are there two actually different applications with the same executable file name, even when considering a number of different

hosts. Nothing prevents two applications to be named similarly in order to cheat the bandwidth allocation scheme taught by Jackowski.

To summarize, in Jackowski, the QoS determination based on the application is made in an external policy server so that the connection from the workstation to a remote server is made first. The actual policing will be performed afterwards by comparing the traffic statistics associated with the application to threshold values. Further, the policy server acquires flow data from the workstations. In Applicant's invention as claimed, the device itself determines an association between the application identifier and a particular QoS policy in a database stored in the device itself, determines in the device the QoS parameters contained in the QoS policy and communicating from the device to the network the QoS parameters to be applied to the at least one of the application and the data stream from or to the application.

The benefit of Applicant's invention according to the recitation in the claims is that it guarantees compatibility to systems where even the initial setting up of a communication path requires knowledge of the QoS. The QoS is required in the communication path establishment procedure. Jackowski is essentially based on statistics collected after the communication path has been used for some time may not be applied to Applicant's invention, because without having the communication path established, no data is sent. Thus, no flow information will ever be received in the edge server.

Thus, claims 1, 4 5 and 7 are not disclosed or suggested by Jackowski. Dependent claims 2, 3, 6 and 8 should be allowable at

least in view of their dependencies. Applicant would appreciate a prompt and thorough response from the Examiner, addressing all of the points identified herein, so that Applicant can properly prepare for an Appeal, should the Examiner not see the patentability of Applicant's invention over the cited art.

For all of the foregoing reasons, it is respectfully submitted that all of the claims now present in the application are clearly novel and patentable over the prior art of record, and are in proper form for allowance. Accordingly, favorable reconsideration and allowance is respectfully requested. Should any unresolved issues remain, the Examiner is invited to call Applicants' attorney at the telephone number indicated below.

The Commissioner is hereby authorized to charge payment for any fees associated with this communication or credit any over payment to Deposit Account No. 16-1350.

Respectfully submitted,

Geza C. Ziegler, Jr.
Reg. No. 44,004
Perman & Green, LLP
425 Post Road
Fairfield, CT 06824
(203) 259-1800 Ext. 134
Customer No.: 2512

10 June 2005
Date

CERTIFICATE OF FACSIMILE TRANSMISSION

I hereby certify that this correspondence is being transmitted by facsimile to (703) 872-9306 the date indicated below, addressed to the Mail Stop AF, Commissioner of Patents, P.O. Box 1450, Alexandria, VA 22313-1450

Date: June 10, 2005

Signature: Meaghan Baye
Person Making Deposit